4. What is the types of SSRF Attacks?

SSRF attacks can be categorized based on their target and how they exploit the server. Let's break down the **different types of SSRF attacks** and how they work:

---

1. Internal SSRF (Internal Resource Access)

- **Description:**
  The attacker forces the server to access **internal resources** that are usually protected by firewalls or other access controls.
- **Goal:**
  To expose or interact with services, files, or applications on internal networks that should be inaccessible from the outside world.
- **Examples:**
  - Accessing a local administrative interface:

    ```
    http://127.0.0.1/admin
    ```

  - Requesting internal services behind the firewall:

    ```
    http://10.0.0.1:8080
    ```

  - Accessing database endpoints or internal APIs that are otherwise not publicly exposed.
- **Impact:**
  The attacker could retrieve sensitive data from internal systems, such as configuration files, private APIs, or admin pages.

---

2. Cloud Metadata SSRF

- **Description:**
  Attackers target **cloud metadata services** that provide sensitive information about the instance, such as credentials, configurations, or access tokens.
- **Goal:**
  To extract metadata from cloud services (AWS, GCP, Azure, etc.) and access sensitive data like API keys, instance metadata, or even IAM roles.
- **Example:**
  - Accessing AWS metadata service (AWS EC2):

    ```
    http://169.254.169.254/latest/meta-data/iam/security-credentials/
    ```

  - GCP metadata service:

    ```
    http://169.254.169.254/computeMetadata/v1/instance/attributes/
    ```

- **Impact:**
  Attackers can steal cloud instance credentials, IAM roles, and other secrets that can escalate their access to other services or instances.

---

3. SSRF for Port Scanning and Enumeration

- **Description:**
  The attacker uses the vulnerable server to perform **port scanning** and **service enumeration** of internal systems that would otherwise be hidden behind firewalls.
- **Goal:**
  To identify open ports and running services on internal networks, which can be used for further exploitation.
- **Example:**
  - Scanning internal IPs for open ports (e.g., port 22 for SSH or 3306 for MySQL):

    ```
    http://127.0.0.1:22
    ```

    ```
    http://10.1.2.3:3306
    ```

- **Impact:**
  Attackers can gather information about internal infrastructure and services, setting the stage for further attacks like **remote code execution** or **exploitation of vulnerable services**.

---

4. SSRF to External Resources (External Resource Fetching)

- **Description:**
  The attacker uses the server to fetch data from external sources, such as third-party APIs, files, or websites, often for malicious purposes.
- **Goal:**
  To retrieve sensitive information or inject malicious payloads into trusted services.
- **Examples:**
  - Fetching an external server that provides malicious content:

    ```
    http://malicious-website.com/malware.zip
    ```

  - Requesting a third-party API with an attacker-controlled request:

    ```
    http://api.some-service.com/get_data?user_id=attacker
    ```

- **Impact:**
  The attacker can get the server to interact with external malicious resources, potentially leading to data leakage, malware infections, or other types of attacks.

---

5. SSRF for Denial of Service (DoS)

- **Description:**
  An attacker exploits SSRF to **overload** an internal resource or service, leading to a **Denial of Service (DoS)** condition.
- **Goal:**
  To cause resource exhaustion by forcing the server to make repeated, resource-heavy requests.
- **Example:**
  - Continuously making requests to internal APIs or services to cause a denial of service:

    ```
    http://localhost:8080/very-heavy-endpoint
    ```
- **Impact:**
  The target server or internal services may crash or become unavailable, affecting the availability of the application or network.

---

6. SSRF for Privilege Escalation

- **Description:**
  Attackers use SSRF to manipulate a vulnerable system and escalate their privileges by interacting with sensitive internal resources.
- **Goal:**
  To access resources that allow the attacker to gain higher privileges within the system.
- **Examples:**
  - Accessing administrative interfaces:

    ```
    http://127.0.0.1/admin
    ```
  - Querying for privileged data on internal services.
- **Impact:**
  This can lead to **unauthorized access**, allowing the attacker to execute commands, access sensitive data, or escalate to a privileged user.

---

7. Blind SSRF

- **Description:**
  In a **blind SSRF**, the attacker does not directly see the response of the request the server makes. Instead, the attacker may infer information based on behaviors or side effects (e.g., server response times or failure conditions).
- **Goal:**
  To infer sensitive data based on server behavior without directly receiving feedback.
- **Example:**
  The attacker sends an SSRF request to an internal API and infers the existence of a service based on response times:

  ```
  http://localhost:80
  ```

- **Impact:**
  While the attacker doesn't see the data directly, they can still gather valuable information that can aid in further attacks.

---

8. SSRF with File Uploads

- **Description:**
  Attackers can leverage SSRF vulnerabilities in file upload mechanisms to fetch files from internal or external resources during the upload process.
- **Goal:**
  To upload files that are malicious or to use SSRF to fetch sensitive files from internal resources and upload them to a public location.
- **Example:**
  An attacker uploads a file with a crafted URL that causes the server to fetch internal files or resources during the upload process:

  ```
  http://internal-server/secretfile.txt
  ```

- **Impact:**
  Sensitive files may be exposed or malicious files uploaded and executed.

Summary of SSRF Attack Types

- **Internal SSRF:** Access private/internal services.
- **Cloud Metadata SSRF:** Extract cloud instance metadata and credentials.
- **Port Scanning SSRF:** Scan internal services and networks.
- **External SSRF:** Interact with external resources or APIs.
- **DoS SSRF:** Overload internal systems or services.
- **Privilege Escalation SSRF:** Gain higher privileges within the network.
- **Blind SSRF:** Infer sensitive data based on server responses.
- **File Upload SSRF:** Exploit file upload mechanisms to access internal resources.
